Ordenación de tuplas para la selección de múltiples e-cursos similares

Cristina Bender, Claudia Deco, María Belén Bernini, Matías Asás

Departamento de Sistemas e Informática, Facultad de Ciencias Exactas, Ingeniería y Agrimenura
Universidad Nacional de Rosario, Argentina
Pellegrini 250, 2000, Rosario, Argentina
{ bender, deco }@fceia.unr.edu.ar

and

Regina Motz
Instituto de Computación, Facultad de Ingeniería,
Universidad de la República, Montevideo, Uruguay
rmotz@fing.edu.uy

Resumen

La búsqueda de cursos a distancia (ó e-cursos) es un recurso cada vez más utilizado por los estudiantes. En este trabajo, se propone una manera de ordenar los resultados de una búsqueda almacenados en una base de datos relacional. La propuesta considera las preferencias y los aspectos culturales del usuario para el ordenado de los resultados. Este ordenamiento consiste en: la conversión de los valores de todos los atributos de los cursos a valores numéricos entre 0 y 1; la generación de una tupla que represente el curso más adecuado para el usuario; y el cálculo de la distancia euclídea entre esta tupla ideal y las tuplas representativas de la características de cada curso. Finalmente, los resultados se ordenan en forma ascendente según esta distancia.

Palabras clave: Búsqueda de cursos, e-cursos

1 Introducción

Los estudiantes poseen distintas preferencias y están inmersos en un entorno cultural. Cuando un usuario obtiene, como resultado de una búsqueda, un conjunto respuesta, algunos de los resultados serán más cercanos a sus preferencias y características culturales que otros. En este trabajo se propone una forma de ordenar estas respuestas en forma descendente en función de la mayor cercanía a estas preferencias. Este trabajo se realiza dentro del marco del Proyecto EduCa [1], cuyo objetivo es el desarrollo de un sistema para la búsqueda de cursos ofrecidos por una federación de universidades, en modalidad a distancia. Esta búsqueda está basada en el uso de ontologías para la recuperación de los cursos de acuerdo a los aspectos culturales y preferencias de los usuarios.

Los aspectos culturales de un usuario son sus preferencias y modos de comportamiento determinados por la cultura del usuario y/o su país de origen. En este proyecto, los aspectos culturales son características que distinguen las preferencias de estudiantes y profesores de distintas regiones. Algunos de los aspectos culturales son: Grado de impaciencia, Actitud, Idioma, Forma de comunicación, Estilos de aprendizaje, Actividades. Todos estos aspectos culturales se especifican en la Ontología de Aspectos Multiculturales [2], que sigue el estándar de metadata de objetos de aprendizaje (LOM: Learning Object Metadata) y usa OWL [3]. La federación de universidades provee un repositorio de cursos enriquecido con metadata. Tanto las preferencias y características culturales de los usuarios, como los metadata de los cursos se almacenan en una base de datos relacional.

Un problema muy importante en la búsqueda de información es el ordenamiento de los resultados obtenidos. En una búsqueda típica, se puede obtener un conjunto grande de documentos resultantes, lo que hace difícil encontrar el que mejor se adecua a las necesidades del usuario. Este problema se hace aún más evidente cuando el objetivo es encontrar un curso específico que satisfaga, además del tema de interés, los conocimientos y características personales y culturales del usuario. Por ejemplo, que el curso esté en un idioma que conozca (y preferentemente maneje fluidamente), que cubra sus condiciones personales y profesionales (holístico visual, grado de impaciencia, actitud), etc. Ya sean características propias de su cultura, de su conocimiento general o preferencias personales, estas condiciones afectan en mayor o menor grado, la importancia y el grado de conformidad que un usuario tendrá con un curso. Estas características son diferentes en cada usuario, y mientras uno puede preferir un cierto curso, otro preferiría uno diferente. Por esto, es importante devolverle a cada usuario los cursos del tema buscado, ordenados de acuerdo a sus preferencias. Si los resultados de una búsqueda no estuvieran ordenados, seguramente el usuario podría elegir alguno de los primeros que mire, y sería muy difícil que encuentre rápidamente el más adecuado de esta forma.

Dado un conjunto de cursos y una consulta de un usuario, el objetivo es encontrar todos los cursos correspondientes al tema de interés del usuario y que se aproximen a su perfil definido por sus preferencias y características culturales. Luego de realizada la búsqueda temática [4] [5], se ordenan las tuplas resultantes de la consulta, para mostrarle al usuario los cursos solicitados en orden descendente, de acuerdo a las preferencias y los aspectos culturales del mismo.

El resto del trabajo se organiza de la siguiente forma, en la Sección 2 se presentan algunos conceptos básicos; en la Sección 3 se muestran trabajos relacionados, en la Sección 4 se describe la metodología propuesta y en la Sección 5 se presenta un caso de uso. Finalmente, en la Sección 6 están las conclusiones.

2 Conceptos Básicos

En el proceso de ordenamiento tratado en este trabajo, se utilizan la lógica difusa (fuzzy logic) y la distancia.

La **lógica difusa** es una rama de la lógica que usa *grados de pertenencia* a conjuntos en lugar de solo dos valores de pertenencia verdadero ó falso. Está fundamentada en la teoría de los Conjuntos Difusos, según la cual el grado de pertenencia de un elemento a un conjunto está determinado por una función de pertenencia, que puede tomar todos los valores reales comprendidos en el intervalo [0,1]. Los esquemas de razonamiento utilizados son esquemas de razonamiento aproximados, que intentan reproducir los esquemas del cerebro humano, donde un objeto puede ser un miembro parcial en un conjunto. Formalmente:

Dado un conjunto X, un conjunto difuso A de X es caracterizado por una función de pertenencia $\mu_A(x)$, que asocia cada elemento x con un grado de pertenencia en A.

$$\mu_A(x) : X \rightarrow [0,1] \qquad \text{donde} \qquad 0 \leq \mu_A(x) \leq 1 \qquad\qquad \forall\, x \in X$$

Entonces, dado el universo X cuyos elementos son $\{x_1, x_2, \ldots x_n\}$, el conjunto difuso A define la función de pertenencia $\mu_A(x)$, que le asigna a cada elemento x_i de X un grado de pertenencia a A entre [0,1].

Una representación gráfica es la siguiente:

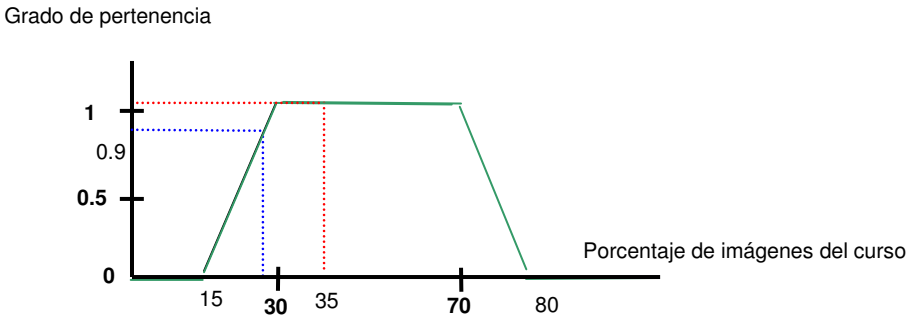

Figura 1: Función de pertenencia al subconjunto difuso *media*

Un curso que contiene entre 30% y 70% de imágenes es un curso con pertenencia de grado 1 al conjunto difuso *mediana cantidad de imágenes*. En cambio, un curso con un 27% de imágenes, tiene un grado de pertenencia de 0,9 al conjunto *mediana cantidad de imágenes*. Cualquier curso con menos de un 15% de imágenes tiene un grado de pertenencia 0 a dicho conjunto.

La lógica difusa puede usarse para representar variables lingüísticas, es decir, variables cuyos valores son palabras o sentencias en lenguaje natural. Cada valor de una variable lingüística se interpreta como un subconjunto difuso de un universo de discurso. La lógica difusa permite representar el significado de modificadores lingüísticos (adverbios) como "muy", "mas o menos", "un poco", etc. Por ejemplo, dado el conjunto difuso A representado por μ_A, "muy A" se representa elevando al cuadrado la función de pertenencia: $(\mu_A)^2$.

Otro concepto utilizado es el de **espacio métrico**. Un espacio métrico es un conjunto de puntos M con una función distancia, ó métrica, asociada d: M x M → ℝ (donde ℝ es el conjunto de los números reales).

Para todo x, y, z en M, esta función debe satisfacer las siguientes condiciones:

$d(x, y) \geq 0$

$d(x, x) = 0$ (reflexividad)

si $d(x, y) = 0$ entonces $x = y$ (identidad de los indiscernibles)

$d(x, y) = d(y, x)$ (simetría)

$d(x, z) \leq d(x, y) + d(y, z)$ (desigualdad triangular).

Existen muchas funciones que cumplen con la definición de función de distancia. La distancia más usual en R^n es la distancia euclídea, que es la que se utiliza en este trabajo:

$$d((x_1, x_2,...,x_n),((y_1, y_2,...,y_n))=((x_1-y_1)^2+(x_2-y_2)^2+...+(x_n-y_n)^2)^{1/2}$$

En particular, la distancia euclídea en R^2 es:

$$d((x_1, x_2),((y_1,y_2))=((x_1-y_1)^2+(x_2-y_2)^2)^{1/2}$$

Otras distancias conocidas en R^2 son:

$d((x_1, x_2),((y_1,y_2))=|x_1-y_1|+|x_2-y_2|$ (City-block distance)

$d((x_1, x_2),((y_1,y_2))=\max\{|x_1-y_1|,|x_2-y_2|\}$ (Chessboard distance)

3 Trabajos relacionados

Existen diversas propuestas para el ordenamiento de resultados de consultas en bases de datos relacionales, pero todavía no hay ninguna que marque un estándar.

En [6] se hace una analogía entre bases de datos relacionales y algoritmos de análisis de links, utilizando grafos de bases de datos para ordenar las tuplas similares a los grafos que se utilizan para ordenar páginas Web. [7] introduce el método SVR (Structured Value Ranking) para ordenar resultados de consultas en bases de datos relacionales muy actualizadas. En [8] se adaptan técnicas de la recuperación de información a bases de datos relacionales. En [9], se presenta un algoritmo de ordenamiento de respuestas a consultas con joins según un puntaje total, el cual se computa combinando los puntajes individuales.

A diferencia de estos trabajos, aquí se propone utilizar técnicas de lógica difusa y calcular las distancias entre tuplas de una tabla de una base de datos relacional. Técnicas como las de [6] no fueron tenidas en cuenta como solución por ser demasiado complejas para lo que este trabajo requiere. La solución de [9], tampoco es válida pues las tuplas a ordenar están todas en la misma relación. Finalmente, se estima que la solución presentada en este trabajo es más adecuada para el tipo de problema presentado aquí que las presentadas en [8], dado que no se adaptarán técnicas de recuperación de información, y en [7], dado que la base de datos usada aquí no es de actualización intensiva.

4 Metodología propuesta

El repositorio de cursos contiene, además de la información sobre su contenido temático, otros metadata tales como idioma, interactividad, y cantidad de imágenes. Dada una consulta de un usuario, el conjunto de cursos resultante que satisface la búsqueda temática, puede ó no satisfacer las preferencias de este usuario.

En este trabajo se propone generar una lista ordenada teniendo en cuenta cuánto se acercan las características de los cursos que cumplen las condiciones temáticas de búsqueda, a las preferencias y características de los usuarios. Así, se tiene una relación *r* con la información de las características descriptivas de todos los cursos que satisfacen la condición temática de búsqueda. Además, a partir de la consulta se genera una relación *user* formada por una única tupla que llamaremos *tupla ideal*, que corresponden al curso ideal según las preferencias y los aspectos culturales de este usuario. Es decir, se tienen las relaciones

r = (IdentificaciónCurso, Característica$_1$, ... , Característica$_n$)

user = (IdentificaciónUser, CaracterísticaUser$_1$, ... , CaracterísticaUser$_n$)

Se calcula la distancia de cada una de las tuplas de *r* con respecto a la tupla ideal, de *user*, que representa las preferencias del usuario, y luego se ordenan estas tuplas en orden ascendente según esta distancia. Aquellos cursos con valores menores en esta distancia aparecen primero y son los que más se acercan a las preferencias y características del usuario.

Es decir, se obtiene una relación *s* de la forma:

s = (IdentificaciónCurso, Distancia)

donde cada tupla v $\in$ *s* es de la forma v = (t [IdentificaciónCurso], d (t, u))

donde t $\in$ *r* , u es la única tupla de *user*, y d es la función elegida para el cálculo de la distancia.

Para el cálculo de estas distancias, se propone primero convertir los valores de los atributos a valores en el rango entre 0 y 1. Los datos a tener en cuenta para el ordenamiento provienen de los aspectos culturales y las preferencias del usuario, además de las propiedades de los cursos. Estos datos tienen atributos de diversos tipos: numérico, carácter, booleano, etc. Algunos de estos atributos pueden contener valores discretos ó rangos. Por ejemplo, no se dice que un determinado curso "tiene práctica" o "no tiene práctica", sino que un curso tiene un determinado porcentaje de práctica. Un porcentaje entre 0% y 30% indica que el curso "tiene poca práctica"; entre 30% y 70% "tiene bastante práctica", y entre 70% y 100% "tiene mucha práctica". Además, la pertenencia de un curso a uno de estos rangos puede ser analizada desde la lógica difusa, es decir considerando grados de pertenencia. Por ejemplo, un curso que tiene 75% de práctica es un curso que "tiene mucha práctica", pero un curso con 69% de práctica, que en la lógica bivaluada no cumpliría el requisito de tener mucha práctica, considerando la lógica difusa se podría pensar que tiene un grado de pertenencia 0,9 a "tiene mucha práctica" y por lo tanto sería importante también recuperarlo. Entonces, la primera decisión de diseño es convertir los valores de los atributos a valores numéricos en el intervalo [0,1]. Para realizar esta conversión se propone una transformación directa para los tipos de datos numéricos, porcentual y booleano, como se muestra en la tabla de la Figura 2.

En la consulta, algunas de las características pueden ser expresadas por el usuario en forma difusa. Por ejemplo, puede decir que le interesan cursos con *mucha práctica.* Entonces, se considera la utilización de lógica difusa, para establecer si el valor de cierto atributo de un curso pertenece al rango pedido por el usuario, o si tiene un grado de pertenencia.

Para el ejemplo de la cantidad de práctica de un curso, los posibles valores son *poca - bastante - mucha*. La función de pertenencia para el rango *mucha* práctica es la que se presenta en la Figura 3.

Tipo de datos	Fórmula para normalizar valores	Ejemplo
Numérico	v / n donde: v : es el valor del atributo n : valor máximo que puede tomar el atributo	Edad
Porcentual	v / 100 donde: v : es el valor del atributo	Porcentaje de práctica que tiene el curso
Booleano	los valores que puede tomar son : True = 1 False = 0	Nacido en Argentina

Figura 2: Conversión de los valores de los atributos

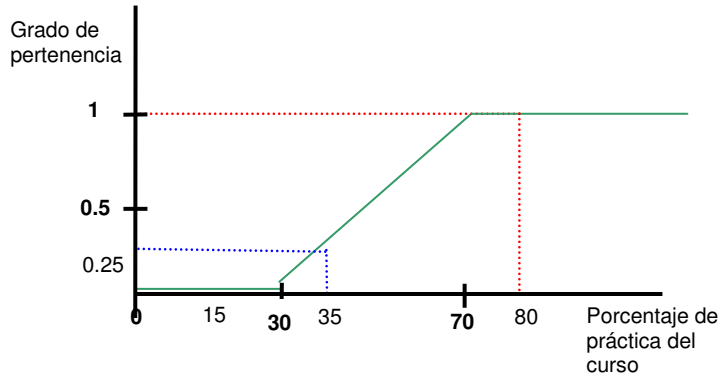

Figura 3: Función de pertenencia al subconjunto difuso *mucha*

Si un curso tiene 35% de práctica, tienen un grado de pertenencia de 0.25 al conjunto *mucha*. Si tiene 80% de práctica el grado de pertenencia es 1, y si tiene menos de 30% es 0.

La cantidad de práctica (poca, media, mucha) que se quiere que tenga el curso buscado, será elegida por el usuario al hacer la búsqueda. Para ordenar los cursos se utiliza la función de pertenencia del conjunto elegido por el usuario para convertir el valor del atributo *cantidad de práctica.* Es decir, si el usuario quiere que el curso tenga mucha práctica, se usará la función *mucha* para encontrar el grado de pertenencia del curso al conjunto *mucha*. Es decir, si un curso encontrado tiene 80% de práctica, el grado de pertenencia para ese curso será de 1. Realizada esta conversión de los atributos, se calculan las distancias de *r* respecto a *user*.

Para el ordenamiento de las tuplas representativas de los cursos, se calcula el peso de cada una de ellas, con respecto a la tupla ideal que representa el perfil del usuario. Para este cálculo se trabaja con varias tuplas simultáneamente. Estas tuplas tienen sus valores convertidos al intervalo [0,1].

En el caso que el usuario no especifique un valor de interés para un atributo, se asigna como valor de dicho atributo el valor de 0.5, que es el valor medio del intervalo [0, 1]. Esta decisión de diseño, fue hecha en base a la suposición de que si el usuario no especifica respuesta puede haber sido porque no entendió la pregunta o porque esa opción no fue de su interés, entre otras posibilidades. No se eligió asignar el valor cero, porque en el procedimiento que aquí se propone se interpretaría como un valor de atributo "indeseable" y no como un valor de atributo "indefinido".

Los atributos de la relación *user* tienen valores que provienen de distintas fuentes: las preferencias indicadas por el usuario en su perfil, y un conjunto de pesos asociados a estos atributos. Además, los atributos de las tuplas correspondientes a los cursos tienen valores que provienen de las

características propias del curso. Cada uno de estos elementos se puede considerar como un vector. El vector de pesos de los atributos, es el que indica qué atributo es más importante que otro, suponiendo que no son de igual importancia todos los atributos. Por ejemplo, para el usuario puede ser más importante que el idioma en que esté el curso sea el de su país de nacimiento, que el hecho que dicho curso tenga muchas figuras. Es decir, le otorga más peso al atributo "idioma materno" que al atributo "holístico visual".

Obtenida la tupla representativa del curso ideal *user* buscado por el usuario, se calculan las distancias entre ésta y cada una de las tuplas de los cursos de la relación *r*, a fin de ordenar obtener la relación *s* con los resultados ordenados en forma creciente de acuerdo a esta distancia.

5 Caso de uso

Supongamos que la información descriptiva de cada curso es la siguiente: idioma (que puede tomar el valor inglés ó español), cantidad de imágenes, cantidad de práctica y cantidad de teoría (todos en valor porcentual), y si el curso es interactivo ó no.

Para una consulta del usuario, se tiene además información de las características del usuario y de sus características culturales: nivel de conocimiento del idioma inglés y nivel de conocimiento del idioma español (ambos en porcentaje), preferencia por cantidad de imágenes, cantidad de práctica y cantidad de teoría (las tres indicadas con *mucha, media* ó *poca*), y preferencia por que el curso sea interactivo ó no lo sea. Además, se tiene información de la importancia ó peso de cada atributo respecto a los otros atributos.

Supongamos un usuario que busca cursos de dibujo, quiere que los cursos tengan gran cantidad de imágenes, que sean interactivos y que tengan mucha práctica, y no especifica nada acerca de la teoría. Dado que es un estudiante de arte sus características son: holístico visual, poco interesado en la teoría y acostumbrado a los cursos interactivos. Este usuario es inglés, y habla medianamente bien el español. Entonces, las preferencias de este usuario son:

Inglés	Español	Cant. imágenes	Cant. práctica	Cant. Teoría	Interactivo
si	**medio**	**mucha**	**mucha**	**null**	**si**

Realizada la conversión de los atributos, según la propuesta presentada en el punto 3, se obtiene la siguiente tupla:

1	**0.6**	**1**	**1**	**0.5**	**1**

Y si el vector de importancia de los atributos es:

1	**0.6**	**0.54**	**0.66**	**0.44**	**0.46**

La tupla del *curso ideal* para el usuario, resulta de promediar los valores componente a componente:

Inglés	Español	Cant. imágenes	Cant. práctica	Cant. Teoría	Interactivo
1	**0.6**	**0.77**	**0.83**	**0.47**	**0.73**

La búsqueda temática se realiza y devuelve cuatro cursos.

Curso	Inglés	Español	Cant. imágenes	Cant. práctica	Cant. Teoría	Interactivo
Curso 1	**1**	**0**	**0.1**	**0.4**	**1**	**0**
Curso 2	**1**	**0**	**1**	**0.5**	**0.5**	**1**
Curso 3	**0**	**1**	**0.5**	**0.5**	**1**	**0**
Curso 4	**0**	**1**	**0.8**	**0.3**	**0.9**	**1**

El primer curso está en inglés, es teórico (tiene más de 80% en teoría), no es interactivo, tiene 40% de práctica, pero pocas imágenes. El segundo curso está en inglés, tiene muchas imágenes, 50% de práctica, 50% de teoría y es interactivo. Y el tercer curso está en castellano, tiene 50% de imágenes, 50% de práctica, es sumamente teórico y no es interactivo. El cuarto curso está en castellano, tiene poca práctica, muchas imágenes, mucha teoría y es interactivo.

Entonces, las distancias con el curso ideal según el usuario son:

Curso	Distancia al curso ideal
Curso 1	$d_1 = ((0)^2 + (0.16)^2 + (0.77\text{-}0.1)^2 + (0.83\text{-}0.4)^2 + (0.47\text{-}1)^2 + (0.73)^2)^{1/2}$ = **1.34**
Curso 2	$d_2 = ((1\text{-}1)^2 + (0.6)^2 + (0.77\text{-}1)^2 + (0.83\text{-}0.5)^2 + (0.47\text{-}0.5)^2 + (0.73\text{-}1)^2)^{1/2}$ = **0.77**
Curso 3	$d_3 = ((1\text{-}0)^2 + (0.6\text{-}1)^2 + (0.77\text{-}0.5)^2 + (0.83\text{-}0.5)^2 + (0.47\text{-}1)^2 + (0.73\text{-}0)^2)^{1/2}$ = **1.47**
Curso 4	$d_4 = ((1\text{-}0)^2 + (0.6\text{-}1)^2 + (0.77\text{-}0.8)^2 + (0.83\text{-}0.3)^2 + (0.47\text{-}0.9)^2 + (0.73\text{-}1)^2)^{1/2}$ = **1.30**

De esta forma, ordenados en forma ascendente en función de esta distancia, el orden de los cursos presentados al usuario es:

Curso 2

Curso 4

Curso 1

Curso 3

Se pueden hacer varias observaciones. En la tupla correspondiente al curso ideal, han quedado tanto el idioma inglés como el idioma español, y ambos con valores altos. Esto siempre quitará valor a los cursos: a los que están en español por tener bajo valor de inglés, y a los que están en inglés por tener bajo valor de español.

Se puede notar además que el Curso 4 se muestra antes que el Curso 1, a pesar que el Curso 1 está

en inglés y el 4 en español. Esto ocurre porque el usuario también habla un poco de español, y el 4 cumple más requisitos que el Curso 1.

6 Conclusión y trabajo futuro

El objetivo de este trabajo es el ordenado de documentos que son devueltos por una búsqueda teniendo en cuenta modificadores tales como las características personales de un usuario y sus aspectos culturales. Para esto, se realiza una conversión tanto de los datos descriptivos de los cursos como de las características personales, los aspectos culturales y las preferencias del usuario, al rango [0, 1]. Luego se generan dos relaciones *user* y *r*. La primera contiene las características que debería tener un curso ideal para un usuario dado. La segunda contiene las tuplas con las características descriptivas de los cursos que cumplen con la condición temática. A continuación, se calculan la distancia euclídea de cada tupla de *r* respecto a la tupla ideal en *user*, obteniendo así una forma sencilla y rápida para ordenar los resultados. Las pruebas de laboratorio realizadas hasta el momento han sido exitosas. Actualmente se está desarrollando un prototipo a fin de realizar una experimentación con un gran número de estudiantes, planteando el uso de condiciones reales, a fin de evaluar esta propuesta.

Referencias

[1] www.fing.edu.uy/inco/grupos/csi/esp/Proyectos/Educa - Red de Educación con Calidad Cultural

[2] Guzmán J. and Motz R. Towards an Adaptive Cultural Web-based Educational System. Proceedings of the 3rd Latin American Web Congress. La Web 2005. IEEE Press. pp 183-186. Buenos Aires, Argentina. 2005

[3] OWL Web Ontology Language Guide http://www.w3.org/TR/owl-guide/

[4] C. Deco, C. Bender, J. Saer, M. Chiari, R. Motz, “Semantic Refinement for Web Information Retrieval. Proceedings of the 3rd Latin American Web Congress. La Web 2005. IEEE Press. pp 106-110. Buenos Aires, Argentina. 2005.

[5] Regina Motz, Jacqueline Guzmán, Claudia Deco, Cristina Bender, “Applying Ontologies to Educational Resources Retrieval driven by Cultural Aspects”. JCS&T Vol. 5 No. 4 December 2005

[6] F. Geerts, H. Mannila, and E. Terzi. Relational link-based ranking. In Proceedings of the International Conference on Very Large Databases (VLDB), August 2004

[7] Lin Guo, Jayavel Shanmugasundaram, Kevin Beyer, Eugene Shekita, "Efficient Inverted Lists and Query Algorithms for Structured Value Ranking in Update-Intensive Relational Databases," icde, pp. 298-309, 21st International Conference on Data Engineering (ICDE'05), 2005.

[8] S. Agrawal, S. Chaudhuri, G. Das, and A. Gionis. Automated ranking of database query results. In Proceedings of the First Biennial Conf. on Innovative Data Systems Research, 2003

[9] Ilyas, I.F., Aref, W.G., Elmagarmid, A.K.: Supporting top-k join queries in relational databases. In: Proceedings of the 29th International Conference on Very Large Databases, Berlin, Germany (2003) 754--765